Inverse regression approach to robust nonlinear high-to-low dimensional mapping

Emeline Perthame, Florence Forbes, Antoine Deleforge

Inverse regression approach to robust nonlinear high-to-low dimensional mapping

Emeline Perthame[a], Florence Forbes[a,*], Antoine Deleforge[b]

[a] *Univ. Grenoble Alpes, Inria, CNRS, LJK, 38000 Grenoble, France*
[b] *Inria, Rennes, France*

Abstract

The goal of this paper is to address the issue of nonlinear regression with outliers, possibly in high dimension, without specifying the form of the link function and under a parametric approach. Nonlinearity is handled via an underlying mixture of affine regressions. Each regression is encoded in a joint multivariate Student distribution on the responses and covariates. This joint modeling allows the use of an inverse regression strategy to handle the high dimensionality of the data, while the heavy tail of the Student distribution limits the contamination by outlying data. The possibility to add a number of latent variables similar to factors to the model further reduces its sensitivity to noise or model misspecification. The mixture model setting has the advantage of providing a natural inference procedure using an EM algorithm. The tractability and flexibility of the algorithm are illustrated on simulations and real high-dimensional data with good performance that compares favorably with other existing methods.

Keywords: EM algorithm, inverse regression, mixture of regressions, nonlinear regression, high dimension, Robust regression, Student distribution.

1. Introduction

A large amount of applications deal with relating explanatory variables (or covariates) to response variables through a regression-type model. In many circumstances, assuming a linear regression model is inadequate and more sensible models are likely to be nonlinear. Other complexity sources include the necessity to take into account a large number of covariates and the possible presence of outliers or influential observations in the data. Estimating a function defined over a large number of covariates is generally difficult because standard regression methods have to estimate a large number of parameters. Then, even in moderate dimension, outliers can result in misleading values for these parameters and predictions may no longer be reliable. In this work, we address these three complication sources by proposing a tractable model that is able to perform nonlinear regression from a high-dimensional space while being robust to outlying data.

A natural approach for modeling nonlinear mappings is to approximate the target relationship by a mixture of linear regression models. Mixture models and paradoxically also the so-called mixture of regression models [10, 17, 20] are mostly used to handle clustering issues and few papers refer to mixture models for actual regression and prediction purposes. Conventional mixtures of regressions are used to add covariates information to clustering models. For high-dimensional data, some penalized approaches of mixtures of regressions have been proposed such as the Lasso regularization [12, 36] but these methods are not designed for prediction and do not deal with outliers. For moderate dimensions, more robust mixtures of regressions have been proposed using Student t distributions [33] possibly combined with trimming [44]. However, in general, conventional mixtures of regressions are inadequate for regression because they assume *assignment independence* [21]. This means that the assignments to each of the regression components are independent of the covariate values. In contrast, in the method we propose, the covariate

*Corresponding author
Email addresses: emeline.perthame@pasteur.fr (Emeline Perthame), florence.forbes@inria.fr (Florence Forbes), antoine.deleforge@inria.fr (Antoine Deleforge)

value is expected to be related to the membership to one of the linear regressions. Each linear regression is mostly active in a specific region of the covariate space.

When extended with assignment dependence, models in the family of mixtures of regressions are more likely to be suitable for regression application. This is the case of the so-called Gaussian Locally Linear Mapping (GLLiM) model [11] that assumes Gaussian noise models and is in its unconstrained version equivalent to a joint Gaussian mixture model (GMM) on both responses and covariates. GLLiM includes a number of other models in the literature. It may be viewed as an affine instance of mixture of experts as formulated in [43] or as a Gaussian cluster-weighted model (CWM) [19] except that the response variable can be multivariate in GLLiM while only scalar in CW models. There have been a number of useful extensions of CW models. The CWt model of [22] deals with non Gaussian distributions and uses Student t distributions for an increased robustness to outliers. The work of [37] uses a factor analyzers approach (CWFA) to deal with CW models when the number of covariates is large. The idea is to overcome the high dimensionality issue by imposing constraints on the covariance matrix of the high-dimensional variable. Incrementally, [38] combines then the Student and Factor analyzers extensions in a so-called CWtFA model. As an alternative to heavy-tailed distributions, some approaches propose to deal with outliers by removing them from the estimation using trimming. Introducing trimming into CWM has then been investigated in [18] but for a small number of covariates and a small number of mixture components. All these CW variants have been designed for clustering and have not been assessed in terms of regression performance.

In contrast, we consider an approach dedicated to regression. To handle the high dimensionality, we adopt an *inverse regression* strategy in the spirit of GLLiM which consists of exchanging the roles of responses and covariates. Doing so, we bypass the difficulty of high-to-low regression by considering the problem the other way around, i.e., low-to-high. We build on the work in [11] by considering mixtures of Student distributions that are able to better handle outliers. As an advantage over the CWtFA approach, our model can deal with response variables of dimension greater than 1. In addition, CWtFA involves the computation of a large empirical covariance matrix of the size of the higher dimension. Furthermore, under our approach, the observed response variables can be augmented with unobserved latent responses. This is interesting for solving regression problems in the presence of data corrupted by irrelevant information for the problem at hand. It has the potential of being well suited in many application scenarios, namely whenever the response variable is only partially observed, because it is neither available, nor observed with appropriate sensors. Moreover, used in combination with the inverse regression trick, the augmentation of the response variables with latent variables acts as a factor analyzer modeling for the noise covariance matrix in the forward regression model. The difference between our approach and CWtFA is further illustrated in Appendix B.

The present paper is organized as follows. The proposed model is presented in Section 2 under the acronym SLLiM for Student Locally Linear Mapping. Its use for prediction is also specified in the same section. Section 3 presents an EM algorithm for the estimation of the model parameters with technical details postponed in Appendix A. Proposals for selecting the number of components and the number of latent responses are described in Section 4. The SLLiM model properties and performance are then illustrated on simulations in Section 5 and real high-dimensional data in Section 6. Section 7 ends the paper with a discussion and some perspectives.

2. Robust mixture of linear regressions in high dimension

We consider the following regression problem. For $n \in \{1, \dots, N\}$, $\mathbf{y}_n \in \mathbb{R}^L$ stands for a vector of response variables with dimension L and $\mathbf{x}_n \in \mathbb{R}^D$ stands for a vector of explanatory variables or covariates with dimension D. These vectors are assumed to be independent realizations of two random variables $\mathbf{Y}$ and $\mathbf{X}$. It is supposed that $L \ll D$ and the number of observations N can be smaller than D. The objective is to estimate the regression function g that we will also call *forward* regression that maps a set of covariates $\mathbf{x}$ to the response variable space, $g(\mathbf{x}) = \mathrm{E}(\mathbf{Y} \mid \mathbf{X} = \mathbf{x})$.

Inverse regression strategy. When the number D of covariates is large, typically more than hundreds, estimating g is difficult because it relies on the exploration of a large dimensional space. A natural approach is therefore to, prior to regression, reduce the dimension of the covariates $\mathbf{x}_1, \dots, \mathbf{x}_N$ and this preferably by taking into account the responses $\mathbf{y}_1, \dots, \mathbf{y}_N$. Methods like partial least squares (PLS), sliced inverse regression (SIR) and Principal component based methods [1, 9, 27, 35, 42] follow this approach, in the category of semi- or nonparametric approaches. When considering parametric models, the issue is usually coming from the necessity to deal with large covariance matrices.

A common solution is then to consider parsimonious modeling of these matrices either by making an oversimplistic independence assumption or using structured parameterization based on eigenvalues decomposition [6] or factor modeling [37]. In this work, we follow a third approach based on the concept of *inverse* regression while remaining parametric as described in [11]. The idea is to bypass the difficulty of estimating a high-to-low dimensional mapping g by estimating instead the other-way-around relationship, namely the low-to-high or *inverse* mapping from $\mathbf{Y}$ to $\mathbf{X}$. This requires then to focus first on a model of the distribution of $\mathbf{X}$ given $\mathbf{Y}$ and implies the definition of a joint model on $(\mathbf{Y}, \mathbf{X})$ to go from one conditional distribution to the other. The reference to a joint distribution is already present in the mixture of experts (MoE) model of [43] in the Gaussian case. However, inversion is not addressed and generally not tractable in non-Gaussian MoE such as those proposed in [8].

Mixture of linear regressions. Because $\mathbf{Y}$ is of moderate dimension, typically less than 10, the inverse regression is likely to be much easier to estimate. However, it is still likely to be nonlinear. An attractive approach for modeling nonlinear data is to use a mixture of linear models; see [11, 19, 43]. Focusing on the modeling of the inverse regression, we consider that each $\mathbf{x}$ is the noisy image of $\mathbf{y}$ obtained from a K-component mixture of affine transformations. This is modeled by introducing the latent variable $Z \in \{1, \ldots, K\}$ such that

$$\mathbf{X} = \sum_{k=1}^{K} \mathbf{1}(Z = k)(\boldsymbol{A}_k \mathbf{Y} + \boldsymbol{b}_k + \boldsymbol{E}_k), \tag{1}$$

where $\mathbf{1}$ is the indicator function, matrix $\boldsymbol{A}_k \in \mathbb{R}^{D \times L}$ and vector $\boldsymbol{b}_k \in \mathbb{R}^D$ define an affine transformation and $\boldsymbol{E}_k \in \mathbb{R}^D$ is an error term not correlated with $\mathbf{Y}$ capturing both the observation noise in $\mathbb{R}^D$ and the reconstruction error due to the affine approximation. Furthermore, $\boldsymbol{E}_k \in \mathbb{R}^D$ is assumed to be zero-mean. To make the affine transformations local, the latent variable Z should depend on $\mathbf{Y}$. For the forward regression $p(\mathbf{Y}|\mathbf{X})$ to be easy to derive from $p(\mathbf{X}|\mathbf{Y})$, it is important to control the nature of the joint $p(\mathbf{Y}, \mathbf{X})$. Once a family of tractable joint distributions is chosen, we can look for one that is compatible with (1). When $\boldsymbol{E}_k$ and $\mathbf{Y}$ are assumed to be Gaussian, such an inverse regression strategy has been proposed and successfully applied to high-dimensional problems in [11] using Gaussian distributions.

Outliers accommodation. The tractability and stability properties of the Gaussian distributions are very convenient and appropriate to the manipulation of conditional and marginal distributions. However, Gaussian models are limited in their ability to handle atypical data due to their short tails. Student t distributions are heavy-tailed alternatives that have the advantage to remain tractable. They have been widely used in robust data analysis including clustering and mixture as already mentioned in the introduction but also linear and nonlinear regressions [29], linear mixed effects models [34], and sample selection models [13, 30]. For a joint model of $\mathbf{Y}$ and $\mathbf{X}$, we therefore consider a mixture of K *generalized* Student distributions. The generalized t version we consider is also referred to as the Arellano-Valle and Bolfarine's Generalized t distribution in Section 5.5 (p. 94) of [25]. The probability density function of an M-dimensional generalized t is given by

$$\begin{aligned} \mathcal{S}_M(\mathbf{x}; \boldsymbol{\mu}, \boldsymbol{\Sigma}, \alpha, \gamma) &= \int_0^\infty \mathcal{N}_M(\mathbf{x}; \boldsymbol{\mu}, \boldsymbol{\Sigma}/u)\, \mathcal{G}(u; \alpha, \gamma)\, \mathrm{d}u \\ &= \frac{\Gamma(\alpha + M/2)}{|\boldsymbol{\Sigma}|^{1/2}\, \Gamma(\alpha)\, (2\pi\gamma)^{M/2}}\, \{1 + \delta(\mathbf{x}, \boldsymbol{\mu}, \boldsymbol{\Sigma})/(2\gamma)\}^{-(\alpha + M/2)}, \end{aligned} \tag{2}$$

where $\mathcal{N}_M(\cdot; \boldsymbol{\mu}, \boldsymbol{\Sigma}/u)$ denotes the M-dimensional Gaussian distribution with mean $\boldsymbol{\mu}$ and covariance $\boldsymbol{\Sigma}/u$ and $\delta(\mathbf{x}, \boldsymbol{\mu}, \boldsymbol{\Sigma}) = (\mathbf{x}-\boldsymbol{\mu})^\top \boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})$ is the square of the Mahalanobis distance between $\mathbf{x}$ and $\boldsymbol{\mu}$. The first order moment exists for $\alpha > 1/2$ and the mean is $\boldsymbol{\mu}$ in this case but $\boldsymbol{\Sigma}$ is not strictly speaking the covariance matrix of the t distribution which is $\gamma\boldsymbol{\Sigma}/(\alpha-1)$ when $\alpha > 1$. When $\alpha = \gamma$, (2) reduces to the standard t distribution. For identifiability reasons, we assume in addition that $\gamma = 1$ as the expression above depends on γ and $\boldsymbol{\Sigma}$ only through the product $\gamma\boldsymbol{\Sigma}$. For $\alpha \neq 1$, the generalized t distribution is therefore different from the standard t distribution. The first equality in (2) shows a useful representation of the distribution as a Gaussian scale mixture which involves an additional Gamma distributed positive scalar latent variable U (the Gamma distribution when the variable is X is denoted by $\mathcal{G}(x; \alpha, \gamma) = x^{\alpha-1}\Gamma(\alpha)^{-1} \exp(-\gamma x)\gamma^\alpha$ where Γ denotes the Gamma function). We found the generalized version easier to manipulate but note that similar developments could have been made with standard t distributions using recent results; see [14] and references therein.

We therefore consider a mixture of K Student distributions with the following $L + D$ dimensional generalized Student distributions, viz.

$$p(\mathbf{Y} = \mathbf{y}, \mathbf{X} = \mathbf{x} \mid Z = k) = \mathcal{S}_{L+D}([\mathbf{y}, \mathbf{x}]^\top; \boldsymbol{m}_k, \boldsymbol{V}_k, \alpha_k, 1) \tag{3}$$

where $[\mathbf{y}, \mathbf{x}]^\top$ denotes the transpose of the vector $[\mathbf{y}, \mathbf{x}]$, $\boldsymbol{m}_k$ is an $L+D$ dimensional mean vector, $\boldsymbol{V}_k$ is a $(D+L)\times(D+L)$ scale matrix and α_k a positive scalar. In applying the inverse regression strategy, the key point is to account for (1) into the parameterization of $\boldsymbol{m}_k$ and $\boldsymbol{V}_k$. Given $Z = k$, it follows from (3) that $\mathbf{Y}$ is Student distributed and $\mathbf{Y}$ can be assumed to have a mean $\boldsymbol{c}_k \in \mathbb{R}^L$ and a scale matrix $\boldsymbol{\Gamma}_k \in \mathbb{R}^{L\times L}$. Then using (1), it comes straightforwardly that

$$\boldsymbol{m}_k = \begin{bmatrix} \boldsymbol{c}_k \\ \boldsymbol{A}_k\boldsymbol{c}_k + \boldsymbol{b}_k \end{bmatrix}, \quad \boldsymbol{V}_k = \begin{bmatrix} \boldsymbol{\Gamma}_k & \boldsymbol{\Gamma}_k\boldsymbol{A}_k^\top \\ \boldsymbol{A}_k\boldsymbol{\Gamma}_k & \boldsymbol{\Sigma}_k + \boldsymbol{A}_k\boldsymbol{\Gamma}_k\boldsymbol{A}_k^\top \end{bmatrix}. \tag{4}$$

The advantage of the joint distribution defined by (3) and (4) is that all conditionals and marginals can be derived and remain Student. More specifically, we obtain the following Student distributions (see Section 5.5, p. 94 of [25]):

$$p(\mathbf{Y} = \mathbf{y} \mid Z = k) = \mathcal{S}_L(\mathbf{y}; \boldsymbol{c}_k, \boldsymbol{\Gamma}_k, \alpha_k, 1), \tag{5}$$

$$p(\mathbf{X} = \mathbf{x} \mid \mathbf{Y} = \mathbf{y}, Z = k) = \mathcal{S}_D(\mathbf{x}; \boldsymbol{A}_k\mathbf{y} + \boldsymbol{b}_k, \boldsymbol{\Sigma}_k, \alpha_k^x, \gamma_k^x), \tag{6}$$

with $\alpha_k^x = \alpha_k + L/2$ and $\gamma_k^x = 1 + \delta(\mathbf{y}, \boldsymbol{c}_k, \boldsymbol{\Gamma}_k)/2$. From (6), it is clear that $\boldsymbol{X}$ is modeled as an affine transformation of $\boldsymbol{Y}$ perturbed by a student error $\boldsymbol{E}_k$ of scale $\boldsymbol{\Sigma}_k$, as defined in (1). From (5), we see that the index k of the affine transformation depends on the location of $\boldsymbol{Y}$.

With no constraints on the parameters, parametrization (4) is general in the sense that all admissible values of $\boldsymbol{m}_k$ and $\boldsymbol{V}_k$ can be written this way; for a proof, see Appendix A of [11] or for $D = 1$ see [22]. Counting the number of parameters, we get $\{D(D-1)+L(L-1)\}/2+DL+D+L$ for the joint model (3), which is not surprisingly symmetric in D and L. The D^2 term is likely to make the model over-parameterized and intractable for large covariate dimensions D. Hence, this inverse regression parameterization only becomes interesting when adding constraints to the parameters. A fundamental constraint that will be used throughout this manuscript is to assume that the scale matrix $\boldsymbol{\Sigma}_k$ of the error term $\boldsymbol{E}_k$ is diagonal. This constraint is quite natural, and corresponds to the assumption that the noise and modeling errors in different entries of $\boldsymbol{Y}$ are uncorrelated. It ensures that all cross-dependencies between the entries of $\boldsymbol{Y}$ are captured by the affine model only, through variable $\boldsymbol{X}$. These cross-dependencies correspond to the term $\boldsymbol{A}_k\boldsymbol{\Gamma}_k\boldsymbol{A}_k$ in (4). The diagonal assumption on $\boldsymbol{\Sigma}_k$ reduces the number of parameters to $\{L(L-1)\}/2 + DL + 2D + L$, removing the D^2 term. Note that proceeding the other way around, i.e., assuming $\boldsymbol{\Gamma}_k$ diagonal instead, would not reduce the number of parameters as drastically. As an example, for $D = 500$ and $L = 2$, the model has 2003 parameters using the inverse strategy and 126,254 using a forward parameterization.

This gain in complexity would not be useful if the forward regression $p(\mathbf{Y} \mid \mathbf{X}, Z = k)$ of interest were not easy to derive. Thankfully, the joint Student model (3) makes it available in closed form. We have:

$$p(\mathbf{X} = \mathbf{x} \mid Z = k) = \mathcal{S}_D(\mathbf{x}; \boldsymbol{c}_k^*, \boldsymbol{\Gamma}_k^*, \alpha_k, 1), \tag{7}$$

$$p(\mathbf{Y} = \mathbf{y} \mid \mathbf{X} = \mathbf{x}, Z = k) = \mathcal{S}_L(\mathbf{y}; \boldsymbol{A}_k^*\mathbf{x} + \boldsymbol{b}_k^*, \boldsymbol{\Sigma}_k^*, \alpha_k^y, \gamma_k^y), \tag{8}$$

with$\alpha_k^y = \alpha_k + D/2$, $\gamma_k^y = 1 + \delta(\mathbf{x}, \boldsymbol{c}_k^*, \boldsymbol{\Gamma}_k^*)/2$.

A new parameterization $\boldsymbol{\theta}^* = \{\boldsymbol{c}_k^*, \boldsymbol{\Gamma}_k^*, \boldsymbol{A}_k^*, \boldsymbol{b}_k^*, \boldsymbol{\Sigma}_k^*\}_{k=1}^K$ is used to illustrate the similarity between Eqs. (5)–(6) and (7)–(8). The parameters $\boldsymbol{\theta}^*$ are easily deduced from $\boldsymbol{\theta}$ as follows:

$$\boldsymbol{c}_k^* = \boldsymbol{A}_k\boldsymbol{c}_k + \boldsymbol{b}_k, \tag{9}$$

$$\boldsymbol{\Gamma}_k^* = \boldsymbol{\Sigma}_k + \boldsymbol{A}_k\boldsymbol{\Gamma}_k\boldsymbol{A}_k^\top, \tag{10}$$

$$\boldsymbol{A}_k^* = \boldsymbol{\Sigma}_k^*\boldsymbol{A}_k^\top\boldsymbol{\Sigma}_k^{-1}, \tag{11}$$

$$\boldsymbol{b}_k^* = \boldsymbol{\Sigma}_k^*(\boldsymbol{\Gamma}_k^{-1}\boldsymbol{c}_k - \boldsymbol{A}_k^\top\boldsymbol{\Sigma}_k^{-1}\boldsymbol{b}_k), \tag{12}$$

$$\boldsymbol{\Sigma}_k^* = (\boldsymbol{\Gamma}_k^{-1} + \boldsymbol{A}_k^\top\boldsymbol{\Sigma}_k^{-1}\boldsymbol{A}_k)^{-1}. \tag{13}$$

It is interesting to consider the structure of the $D \times D$ scale matrix $\boldsymbol{\Gamma}_k^*$ of $(\mathbf{X}|Z)$ in (10). Since $\boldsymbol{\Sigma}_k$ is assumed diagonal, we obtain a factor analyzer structure of $\boldsymbol{\Gamma}_k^*$. This factor decomposition shows some similarity with the

cluster-weighted modeling approach of [38] which assumes a factor decomposition of the high-dimensional covariates covariance matrix in a forward modeling. However some differences can be pointed out. First, our parameterization is more parsimonious with qD less parameters if q is the number of factors used in [38]. Then, the joint model used in [38] — see their Eq. (4) — is not a joint Student model because the degrees of freedom parameters in their joint probability density function decomposition are the same for the conditional and marginal pdf. Typically a joint Student model would imply instead a degree of freedom parameter that depends on $\mathbf{x}$ in the expression of $p(\mathbf{y} \mid \mathbf{x})$. One consequence of that is that [38] cannot use a regular EM for parameter estimation but have to use an AECM algorithm [31]. In terms of performance, we could not really assess the performance of their approach on very high-dimensional data as the code of this recent work is not available. However for comparison, we applied our method on the two real data sets used in [38] for which $L = 1$ and $D = 6$ and 13, respectively. The results are reported in Appendix B and confirm that the two models yield similar results on simple well separated cluster examples and different results on more complex data.

Low-to-high mapping and prediction. Defining π_k as the probability $\Pr(Z = k)$, Eqs. (5)–(13) show that the whole model is entirely defined by a set of parameters denoted by $\boldsymbol{\theta} = \{\boldsymbol{c}_k, \boldsymbol{\Gamma}_k, \boldsymbol{A}_k, \boldsymbol{b}_k, \boldsymbol{\Sigma}_k, \alpha_k, \pi_k\}_{k=1}^K$ and an inverse regression from $\mathbb{R}^L$ (low-dimensional space) to $\mathbb{R}^D$ (high-dimensional space) can be obtained using the following *inverse conditional density*:

$$p(\mathbf{X} = \mathbf{x} \mid \mathbf{Y} = \mathbf{y}; \boldsymbol{\theta}) = \sum_{k=1}^{K} \frac{\pi_k \mathcal{S}_L(\mathbf{y}; \boldsymbol{c}_k, \boldsymbol{\Gamma}_k, \alpha_k, 1)}{\sum_{j=1}^{K} \pi_j \mathcal{S}_M(\mathbf{y}; \boldsymbol{c}_j, \boldsymbol{\Gamma}_j, \alpha_j, 1)} \mathcal{S}_D(\mathbf{x}; \boldsymbol{A}_k \mathbf{y} + \boldsymbol{b}_k, \boldsymbol{\Sigma}_k, \alpha_k^x, \gamma_k^x).$$

Also, more importantly, the forward regression of interest, i.e., from $\mathbb{R}^D$ (the high dimension) to $\mathbb{R}^L$ (the low dimension), is obtained from the *forward conditional density*:

$$p(\mathbf{Y} = \mathbf{y} \mid \mathbf{X} = \mathbf{x}; \boldsymbol{\theta}^*) = \sum_{k=1}^{K} \frac{\pi_k \mathcal{S}_D(\mathbf{x}; \boldsymbol{c}_k^*, \boldsymbol{\Gamma}_k^*, \alpha_k, 1)}{\sum_{j=1}^{K} \pi_j \mathcal{S}_D(\mathbf{x}; \boldsymbol{c}_j^*.\boldsymbol{\Gamma}_j^*, \alpha_j, 1)} \mathcal{S}_L(\mathbf{y}; \boldsymbol{A}_k^* \mathbf{x} + \boldsymbol{b}_k^*, \boldsymbol{\Sigma}_k^*, \alpha_k^y, \gamma_k^y). \tag{14}$$

The latter involves parameters $\{\pi_k, \alpha_k\}_{k=1}^K$ and the *forward regression* parameters $\boldsymbol{\theta}^* = \{\boldsymbol{c}_k^*, \boldsymbol{\Gamma}_k^*, \boldsymbol{A}_k^*, \boldsymbol{b}_k^*, \boldsymbol{\Sigma}_k^*\}_{k=1}^K$ that can be analytically derived from the *inverse regression parameters* $\boldsymbol{\theta}$ with a drastic reduction of the model size, making tractable its estimation. Indeed, if we consider isotropic equal $\boldsymbol{\Sigma}_k$, the dimension of the learned parameter vector $\boldsymbol{\theta}$ is $O(DL + L^2)$, while it would be $O(DL + D^2)$ using a forward model (recall that $L \ll D$).

Then, when required, a prediction of response $\mathbf{y}$ corresponding to an input $\mathbf{x}$ can be proposed using the expectation of $p(\mathbf{Y} = \mathbf{y}|\mathbf{X} = \mathbf{x}; \boldsymbol{\theta})$ in (14) that can be obtained using:

$$\mathrm{E}(\mathbf{Y} \mid \mathbf{x}; \boldsymbol{\theta}^*) = \sum_{k=1}^{K} \frac{\pi_k \mathcal{S}_D(\mathbf{x}; \boldsymbol{c}_k^*, \boldsymbol{\Gamma}_k^*, \alpha_k, 1)}{\sum_{j=1}^{K} \pi_j \mathcal{S}_D(\mathbf{x}; \boldsymbol{c}_j^*, \boldsymbol{\Gamma}_j^*, \alpha_j, 1)} (\boldsymbol{A}_k^* \mathbf{x} + \boldsymbol{b}_k^*).$$

Response augmentation. In some applications, it is common that some additional factors interfere with the responses without being measured. For instance in the field of sound-source localization, the acoustic input depends on both the source position, which can be observed and of reverberations, that are strongly dependent on the experimental conditions, and for which ground-truth data are barely available. See [11] for other examples. This phenomenon can be modeled by assuming that the response $\mathbf{Y}$ is partially observed. The $\mathbf{Y}$ vector is therefore decomposed as

$$\mathbf{Y} = \begin{bmatrix} \mathbf{T} \\ \mathbf{W} \end{bmatrix},$$

where $\mathbf{T} \in \mathbb{R}^{L_t}$ is the observed part and $\mathbf{W} \in \mathbb{R}^{L_w}$ is not observed and is considered as latent. Accordingly, the dimension of the response is $L = L_t + L_w$, where L_t is the number of observed responses and L_w is the number of unobserved factors. To account for this decomposition, we introduce the notations below

$$\boldsymbol{c}_k = \begin{bmatrix} \boldsymbol{c}_k^t \\ \boldsymbol{c}_k^w \end{bmatrix}, \quad \boldsymbol{\Gamma}_k = \begin{bmatrix} \boldsymbol{\Gamma}_k^t & \mathbf{0} \\ \mathbf{0} & \boldsymbol{\Gamma}_k^w \end{bmatrix} \quad \text{and} \quad \boldsymbol{A}_k = [\boldsymbol{A}_k^t, \boldsymbol{A}_k^w]$$

with $\boldsymbol{A}_k^t$ (respectively $\boldsymbol{A}_k^w$) a $p \times L_t$ (respectively $p \times L_w$) matrix. For identifiability, $\boldsymbol{c}_k^w$ and $\boldsymbol{\Gamma}_k^w$ must be fixed and are usually set to $\boldsymbol{c}_k^w = \boldsymbol{0}$ and $\boldsymbol{\Gamma}_k^w = \mathbb{I}_{L_w}$ ($\mathbb{I}_M$ denotes the $M \times M$ identity matrix). Interestingly, the introduction of an L_w-dimensional latent response variable $\boldsymbol{W}$ brings more flexibility to the model by allowing the modeling of more complex dependencies across entries of $\boldsymbol{X}$. More specifically, Eq. (10) $\boldsymbol{\Gamma}_k^* = \boldsymbol{\Sigma}_k + \boldsymbol{A}_k\boldsymbol{\Gamma}_k\boldsymbol{A}_k^\top$ illustrates that dependencies across entries of $\boldsymbol{X}$ are decomposed into a diagonal part $\boldsymbol{\Sigma}_k$, which is the variance of the noise of the inverse regression, with independent entries, and a low rank part $\boldsymbol{A}_k\boldsymbol{\Gamma}_k\boldsymbol{A}_k^\top$, which models dependencies across entries of $\boldsymbol{X}$. Augmenting the response, meaning adding latent factors $\boldsymbol{W}$ allows to increase the rank of the low-rank part $\boldsymbol{A}_k\boldsymbol{\Gamma}_k\boldsymbol{A}_k^\top$ from $L = L_t$ to $L = L_t + L_w$ and therefore to model more complex correlations among covariates. This augmented model is the one we refer to as Student Locally Linear Mapping (SLLiM) in the following of the paper.

Remark. Considering the addition of these factors $\mathbf{W}$, one can show that the conditional distribution function of $\mathbf{X}$ given the observed $\mathbf{T}$ is

$$p(\mathbf{X} = \mathbf{x} \mid \mathbf{T} = \mathbf{t}, Z = k) = \mathcal{S}_D(\mathbf{x}, \boldsymbol{A}_k[\mathbf{t}, \boldsymbol{c}_k^w]^\top + \boldsymbol{b}_k, \boldsymbol{\Sigma}_k + \boldsymbol{A}_k^w\boldsymbol{\Gamma}_k^w\boldsymbol{A}_k^{w\top}, \alpha_k^y, \gamma_k^y).$$

Therefore, compared to the non-augmented version of our model, the high-dimensional matrix $\boldsymbol{\Sigma}_k$ is replaced by $\boldsymbol{\Sigma}_k + \boldsymbol{A}_k^w\boldsymbol{\Gamma}_k^w\boldsymbol{A}_k^{w\top}$, which corresponds to a factor model with L_w factors. This allows for more general dependence structures while remaining tractable in high dimension. Similarly in the forward model, $\boldsymbol{\Gamma}_k^* = \boldsymbol{\Sigma}_k + \boldsymbol{A}_k^w\boldsymbol{\Gamma}_k^w\boldsymbol{A}_k^{w\top} + \boldsymbol{A}_k^t\boldsymbol{\Gamma}_k^t\boldsymbol{A}_k^{t\top}$ is also augmented with a L_w factor structure.

3. Estimation procedure

In contrast to the Gaussian case, no closed form solution exists for the maximum likelihood estimation of the parameters for a Student t distribution but tractability is maintained, both in the univariate and multivariate case, via the Gaussian scale mixture representation introduced in Eq. (2) [2, 5, 33]. A closed form EM algorithm can be used to estimate the parameters $\boldsymbol{\theta} = \{\boldsymbol{c}_k, \boldsymbol{\Gamma}_k, \boldsymbol{A}_k, \boldsymbol{b}_k, \boldsymbol{\Sigma}_k\, \pi_k, \alpha_k\}_{k=1}^K$. The EM principle is based on a data augmentation strategy that consists of augmenting the observed data $(\mathbf{x}, \mathbf{t})_{1:N} = \{\mathbf{x}_n, \mathbf{t}_n\}_{n=1}^N$ with missing variables. We will write $(\,)_{1:N}$ to indicate that an N-sample of the argument is considered. In our case, there are three sets of missing variables $(Z)_{1:N}, (\mathbf{W})_{1:N}$ and $(U)_{1:N}$ coming from the Gaussian scale representation (2). Parameters are then updated by iteratively maximizing the conditional expectation of the complete data log-likelihood, given the observed training data $(\mathbf{x}, \mathbf{t})_{1:N} = \{\mathbf{x}_n, \mathbf{t}_n\}_{n=1}^N$, and the last update $\boldsymbol{\theta}^{(i)}$. At iteration $i + 1$, we look for the new set $\boldsymbol{\theta}^{(i+1)}$ that verifies

$$\boldsymbol{\theta}^{(i+1)} = \arg\max_{\boldsymbol{\theta}} \mathrm{E}[\ln p\{(\mathbf{x}, \mathbf{t}, \mathbf{W}, U, Z)_{1:N}; \boldsymbol{\theta}\} \mid (\mathbf{x}, \mathbf{t})_{1:N}; \boldsymbol{\theta}^{(i)}],$$

where uppercase letters indicate random variables with respect to which the expectation is computed. Using that responses $\mathbf{T}$ and latent variables $\mathbf{W}$ are independent given hidden variables Z and U and that $\boldsymbol{c}_k^w$ and $\boldsymbol{\Gamma}_k^w$ are fixed, the expected log likelihood to be maximized splits into two parts. The conditional distribution of $(\mathbf{W}, U, Z)_{1:N}$ is decomposed into three distributions denoted by $\widetilde{r}_Z$, $\widetilde{r}_{U|Z}$ and $\widetilde{r}_{W|Z,U}$ with $\widetilde{r}_Z = p\{(Z)_{1:N} \mid (\mathbf{x}, \mathbf{t})_{1:N}; \boldsymbol{\theta}^{(i)}\}$, $\widetilde{r}_{U|Z} = p\{(U)_{1:N} \mid (\mathbf{x}, \mathbf{t}, Z)_{1:N}; \boldsymbol{\theta}^{(i)}\}$ and $\widetilde{r}_{W|Z,U} = p\{(\mathbf{W})_{1:N} \mid (\mathbf{x}, \mathbf{t}, Z, U)_{1:N}; \boldsymbol{\theta}^{(i)}\}$. One then gets the following decomposition of the conditional expectation of the complete log-likelihood,

$$\mathrm{E}_{\widetilde{r}_Z}\mathrm{E}_{\widetilde{r}_{U|Z}}\mathrm{E}_{\widetilde{r}_{W|Z,U}}[\ln p\{(\mathbf{x})_{1:N}|(\mathbf{t}, \mathbf{W}, U, Z)_{1:N}; \boldsymbol{\theta}\}] + \mathrm{E}_{\widetilde{r}_Z}\mathrm{E}_{\widetilde{r}_{U|Z}}[\ln p\{(\mathbf{t}, U, Z)_{1:N}; \boldsymbol{\theta}\}]. \tag{15}$$

The proposed EM algorithm computes and maximizes iteratively this quantity and iterates respectively over E and M steps detailed in Appendix Appendix A.

4. Model selection issues

The SLLiM model relies on the preliminary choice of two numbers, the number K of linear regressions and the number L_w of additional latent variables. We mention below simple ways to select such values. A more thorough study of this issue would be useful but it is out of the scope of the present paper.

Determining the number of clusters K. This number can be equivalently interpreted as the number of affine regressions or as the number of mixture components. In this latter case, it is known that regularity conditions do not hold for the chi-squared approximation used in the likelihood ratio test statistic to be valid. As an alternative, penalized likelihood criteria like the Bayesian Information Criterion (BIC) are often used for clustering issues because interpretation of the results may strongly depend on the number of clusters. In our regression context, the specific value of K may be less important. For SLLiM, K can be set to an arbitrary value large enough to catch nonlinear relationships in a D-dimensional space, while being vigilant that the number of observations is large enough to allow a stable fit of all K components. In practice, we will compare this solution to the one using BIC to select K.

Determining the number of latent variables L_w. The selection of L_w is similar to the issue of selecting the number of factors in a factor analyzer model [3]. Regularity conditions usually hold for tests on the number of factors but as in [3], we rather investigate the use of BIC for choosing L_w. When K is not fixed, BIC can be computed for varying couples (K, L_w) but the available sample size usually limits the range of values that can be tested with reliable BIC computation. For this reason, if necessary we will rather fix K to some value not too large so as to be able to investigate a larger range of L_w values.

5. Simulation study

The code for SLLiM is available as an R package called `xLLiM` on the CRAN at

`https://cran.r-project.org/web/packages/xLLiM/`.

We propose to assess the prediction accuracy and the robustness of the proposed method through an intensive simulation study.

Simulation setting. In this simulation study, we build on the simulation plan described in [11]. We propose a simulation design based on the same approach, except that the purpose of this simulation section is to study the prediction accuracy of our method in presence of extreme values, heavy-tail distributions and non symmetric noises. To do so, we generate data through a wide range of non-Gaussian distributions.

Responses and covariates are generated according to an inverse regression model. This approach allows to generate high-dimensional covariates, dimension by dimension from a small number of variables (responses). Therefore, from a vector of responses $\mathbf{y}$, a vector of covariates $\mathbf{x}$ is generated using the following model:

$$\mathbf{x} = \boldsymbol{f}(\mathbf{y}) + \boldsymbol{e} = \boldsymbol{f}(t, \mathbf{w}) + \boldsymbol{e}, \tag{16}$$

where $\boldsymbol{e}$ is an error term with distribution described hereafter, and $\boldsymbol{f}$ is one of the three nonlinear regression functions specified below. We consider three regression functions $\boldsymbol{f} = (f_1, \ldots, f_d)$, $\boldsymbol{g} = (g_1, \ldots, g_d)$ and $\boldsymbol{h} = (h_1, \ldots, h_d)$ of the form:

$$\begin{aligned}
f_d(\mathbf{y}) &= f_d(t, w_1) = \alpha_d \cos(\eta_d t/10 + \phi_d) + \gamma_d w_1^3, \\
g_d(\mathbf{y}) &= g_d(t, w_1) = \alpha_d \cos(\eta_d t/10 + \beta_d w_1 + \phi_d), \\
h_d(\mathbf{y}) &= h_d(t, w_1, w_2) = \alpha_d \cos(\eta_d t/10 + \beta_d w_1 + \phi_d) + \gamma_d w_2^3.
\end{aligned}$$

The regression parameters are uniformly sampled as follows: $\alpha_d \in [0, 2]$, $\eta_d \in [0, 4\pi]$, $\phi_d \in [0, 2\pi]$, $\beta_d \in [0, \pi]$, $\gamma_d \in [0, 2]$. Notice that each function depends on an observed response denoted by t and one or two unobserved factors denoted by (w_1, w_2), whose values are known and necessary to simulate the covariates but they are supposed not to be observed for the evaluation. The goal is to assess the impact of non-observed responses on prediction performance for the compared methods. In this simulation setting, t is uniformly sampled in $[0, 10]$ and $\boldsymbol{w} = (w_1, w_2)$ in $[-1, 1]^2$. The covariate dimension is set to $D = 50$ and training and testing samples are generated with $N = 200$ observations.

Then, the following distributions for the error term $\boldsymbol{e}$ are considered: standard Gaussian, standard Student with $\alpha = 2$ and $\gamma = 1$, centered log-normal $\mathcal{LN}(0, 1)$, centered Cauchy with scale parameter set to 100 (which generates extreme values) and Uniform $\mathcal{U}(-\sqrt{3}, \sqrt{3})$. The error terms are then normalized in order to reach an average SNR (Signal to Noise Ratio) around 5, which is close to the SNR of the simulation plan in [11]. As proposed in [12], we define the SNR for dimension d by $\mathrm{SNR}_d = \mathrm{var}(x_d)/\mathrm{var}(e_d)$, where x_d and e_d are respectively the dth component of $\mathbf{x}$ and $\boldsymbol{e}$, and we deduce the global SNR by $\mathrm{SNR} = \sum_{d=1}^{D} \mathrm{SNR}_d/D$. This simulation plan is run 100 times.

Results. The following prediction methods are compared:

a) Two versions of the proposed model of robust nonlinear regression (SLLiM). In the first version (denoted by SLLiM-0), the number of latent factors L_w is set to 0 and the number of clusters K is estimated using BIC. In a second version (denoted by SLLiM), both K and L_w are estimated using BIC.

b) A Gaussian version of our model (GLLiM [11]) using the Matlab toolbox available at `http://team.inria.fr/perception/gllim_toolbox/`. The numbers K and L_w are chosen as above and the methods are referred to GLLiM-0 and GLLiM in the results.

c) Random forests [7] performed using the default options of the R package `randomForest`.

d) Multivariate Adaptive Regression Splines [16] using the `mars` function of the `mda` R package.

e) Support Vector Machine (SVM [40]) performed with several kernels (linear, Gaussian and polynomial) as in the R package `e1071` [24].

f) Sliced Inverse Regression (SIR [27]) followed by a polynomial regression of degree 3 performed on the SIR components. We assess predictions with 1 to 10 directions, using the `dr` function of the `dr` R package which implements dimension reduction methods.

g) Relevant Vector Machine (RVM) which is known to perform better in some cases than SVM [39]. We use the Matlab code at `http://mi.eng.cam.ac.uk/ at315/MVRVM.htm`). We compare results achieved by several kernels (Gaussian, linear and Cauchy which is a heavy-tailed kernel) for 60 values of the scale parameter from 0.1 to 6 with a 0.1 increment.

For SIR, SVM and RVM, we present the results corresponding to parameter values leading to the best average prediction, meaning 4 or 5 directions for SIR depending on design and Gaussian kernel for both SVM and RVM. The scale parameter for Gaussian kernel in RVM is set to 0.15 for $\boldsymbol{f}$ and $\boldsymbol{g}$ regression functions, and to 0.12 for $\boldsymbol{h}$. Note that contrary to SVM, RVM and SIR, which require a careful choice of kernel or regression function, GLLiM and SLLiM are entirely parameter-free thanks to the use of BIC. Results of the simulation study are presented in Table 1. The prediction error is assessed using the normalized root mean squared error (NRMSE) defined as

$$\text{NRMSE} = \left\{ \sum_i (t_i - \hat{t}_i)^2 / \sum_i (t_i - \bar{t}_{\text{train}})^2 \right\}^{1/2}.$$

The NRMSE is a normalized version of the RMSE in which we compare the prediction rate to the one reached by predicting all responses by the mean of the training responses, independently of the covariates. A NRMSE equal to 1 means that the method performs as well as one that would set all predictions to the training responses mean. The smaller the NRMSE the better.

Table 1 displays the NRMSE achieved by each method, for each distribution of the noise in model (16). Regarding mixture-based methods, SLLiM and SLLiM-0 improve its Gaussian counterparts GLLiM-0 and GLLiM, in both situations with and without latent factors. Unsurprisingly, adding latent factors in SLLiM leads to better predictions, as the simulation plan includes hidden responses. Regardless of the noise distribution, SLLiM is competitive in most situations. More specifically, the contribution of SLLiM is all the more interesting as the error distribution is different from the Gaussian distribution and heavy tailed.

6. Experiments on real high-dimensional data

In this section, the performance of the proposed method is assessed on two datasets. The following subsection illustrates the properties of the model on data with a small number of observations with respect to the number of variables and the second subsection investigates the contribution of SLLiM over the Gaussian version (GLLiM) on a dataset for which GLLiM is already performing well.

Table 1: High-dimensional simulation study ($D = 50$): Average NRMSE and standard deviations computed over 100 runs. Best values are indicated in bold chatacters.

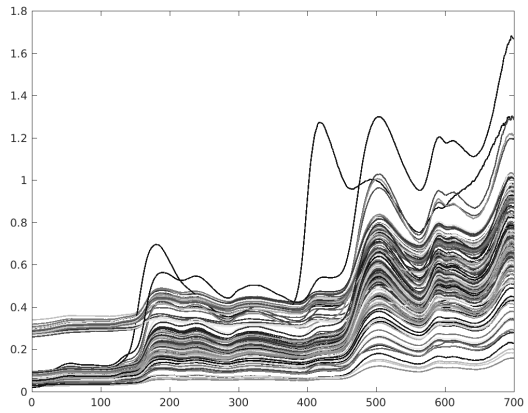
	Gaussian			Student		
	f	g	h	f	g	h
GLLiM-0	0.175 (0.083)	0.245 (0.071)	0.421 (0.097)	0.223 (0.094)	0.301 (0.091)	0.468 (0.100)
SLLiM-0	0.163 (0.076)	0.272 (0.084)	0.372 (0.097)	0.172 (0.092)	0.304 (0.107)	0.401 (0.113)
GLLiM	0.129 (0.056)	0.207 (0.068)	0.259 (0.086)	0.130 (0.057)	0.234 (0.084)	0.297 (0.092)
SLLiM	**0.078** (0.035)	**0.173** (0.056)	**0.235** (0.075)	**0.085** (0.037)	**0.183** (0.059)	**0.252** (0.097)
Random Forest	0.182 (0.045)	0.289 (0.075)	0.352 (0.073)	0.185 (0.045)	0.302 (0.065)	0.355 (0.073)
MARS	0.261 (0.047)	0.468 (0.123)	0.507 (0.134)	0.271 (0.057)	0.525 (0.138)	0.551 (0.158)
SVM	0.214 (0.034)	0.283 (0.040)	0.326 (0.052)	0.275 (0.034)	0.339 (0.040)	0.356 (0.039)
SIR	0.200 (0.036)	0.398 (0.098)	0.414 (0.103)	0.225 (0.045)	0.442 (0.106)	0.430 (0.100)
RVM	0.180 (0.035)	0.224 (0.029)	0.269 (0.044)	0.320 (0.053)	0.334 (0.055)	0.340 (0.049)
	Log-normal			Cauchy		
	f	g	h	f	g	h
GLLiM-0	0.202 (0.090)	0.275 (0.079)	0.452 (0.107)	0.281 (0.130)	0.361 (0.099)	0.513 (0.123)
SLLiM-0	0.173 (0.082)	0.271 (0.079)	0.389 (0.109)	0.164 (0.083)	0.314 (0.100)	0.408 (0.110)
GLLiM	0.119 (0.056)	0.229 (0.088)	0.301 (0.104)	0.231 (0.100)	0.338 (0.126)	0.358 (0.114)
SLLiM	**0.088** (0.039)	**0.180** (0.058)	**0.250** (0.088)	**0.100** (0.041)	0.208 (0.078)	0.281 (0.101)
Random Forest	0.161 (0.035)	0.239 (0.054)	0.323 (0.068)	0.123 (0.039)	**0.182** (0.042)	**0.270** (0.062)
MARS	0.244 (0.057)	0.412 (0.119)	0.488 (0.121)	0.767 (0.980)	1.323 (1.053)	1.029 (0.952)
SVM	0.274 (0.036)	0.336 (0.040)	0.358 (0.045)	0.334 (0.037)	0.373 (0.041)	0.377 (0.037)
SIR	0.223 (0.045)	0.423 (0.097)	0.437 (0.103)	0.223 (0.090)	0.426 (0.125	0.444 (0.109)
RVM	0.280 (0.054)	0.279 (0.039)	0.317 (0.039)	0.459 (0.088)	0.451 (0.074)	0.411 (0.054)
	Uniform					
	f	g	h			
GLLiM-0	0.166 (0.081)	0.272 (0.074)	0.445 (0.115)			
SLLiM-0	0.166 (0.084)	0.281 (0.075)	0.388 (0.094)			
GLLiM	0.131 (0.058)	0.217 (0.073)	0.264 (0.089)			
SLLiM	**0.078** (0.029)	**0.184** (0.065)	**0.231** (0.083)			
Random Forest	0.167 (0.042)	0.288 (0.066)	0.351 (0.078)			
MARS	0.260 (0.049)	0.490 (0.120)	0.511 (0.111)			
SVM	0.206 (0.033)	0.278 (0.042)	0.334 (0.046)			
SIR	0.196 (0.047)	0.420 (0.095)	0.410 (0.087)			
RVM	0.170 (0.029)	0.221 (0.027)	0.276 (0.037)			

6.1. Orange juice dataset

The proposed method is now applied to the Orange juice public dataset in order to illustrate that SLLiM is competitive in high-dimensional settings with $D \approx N$. The goal is to assess the efficiency of SLLiM in such a setting and to illustrate that latent factors $\mathbf{W}$ introduced in the model are useful to catch dependency among features.

Data. The data contains near-infrared spectra measured on $N = 218$ orange juices. It can be downloaded at `http://www.ucl.ac.be/mlg/index.php?page=DataBases` or from the open-source `cggd` R package available on the CRAN in the object `data(OJ)`. The length of each spectrum is 700 and the aim is to model the relationship between the level of sucrose ($L = 1$) and the spectra. Figure 1 shows the N spectra. The curves are quite similar, even if some spectra appear to have extreme values and exhibit isolated peaks.

Figure 1: Curves of orange juice spectra



Method. First, spectra are decomposed on a splines basis using the `smooth.splines` function of the R software. $D = 134$ knots are retained. Reducing the data to the splines coefficients makes the variables exchangeable and is also convenient to reduce the dimension while preserving information. In this section, we compare the same methods as in the simulation study of Section 5 except that we added "GLLiM (K=10)" and "SLLiM (K=10)" in which the number of clusters is arbitrarily set to $K = 10$, which is large enough to catch nonlinear relationships regarding to the dimension of the data. The prediction accuracy is evaluated using a Leave-One-Out cross-validation (LOO-CV). The model is estimated on training data sets of size 217 and a normalized prediction error is computed by predicting the sucrose level of the single remaining observation. Each method is therefore assessed 218 times. As the number of observations is small regarding to the number of variables, the presence of outliers in the testing dataset in the CV generates artificially bad predictions which are not absorbed by the size of the testing sample. For these reasons, the computed NRMSE is affected by outliers: large prediction errors are observed on outlying data points. We therefore compute the median instead of the mean of the NRMSE values to get a better insight on the respective methods prediction performance.

Results. Table 2 shows the median of the NRMSE and percentage of outliers for the compared methods. Outliers are defined as runs leading to an error greater than the error obtained using the training data set mean as predictor. Results are presented with parameters values leading to the best results, namely linear kernel for SVM, one direction for SIR and Gaussian kernel with scale parameter set to 0.70 for RVM. For both GLLiM and SLLiM, setting the number of clusters to 10 and choosing the number of latent factors with BIC leads to the best prediction. For SLLiM, BIC criterion retained 9 to 12 latent factors for 91% of the CV-runs (similar proportions for GLLiM). Selecting K by BIC leads to values between 8 and 10 for 96 % of the CV-runs. For the two different ways to select K and L_w, SLLiM always outperforms its Gaussian counterpart. In Table 2, GLLiM-0 (resp. SLLiM-0) denotes the results for $K = 10$ and $L_w = 0$. It shows worse predictions for both GLLiM and SLLiM and illustrates the advantage of adding latent

Table 2: LOO-CV results for Orange juice data after decomposition on splines. Median NRMSE and % of outliers in parenthesis.

Procedure	Median NRMSE (% outliers)
SLLiM (BIC)	**0.420 (22.93)**
SLLiM (K=10)	**0.388 (27.06)**
SLLiM–0 (K=10)	0.885 (45.87)
GLLiM (BIC)	0.623 (34.86)
GLLiM (K=10)	0.466 (29.36)
GLLiM–0 (K=10)	1.022 (50.46)
Random forests	0.589 (31.19)
MARS	0.629 (33.03)
SVM	**0.425 (24.77)**
SIR	1.020 (51.83)
RVM	0.536 (33.49)

factors to the model. Results for $L_w = 0$ and K selected by BIC are not presented but are similar. Among the other compared methods, the best prediction error is obtained using SVM with a linear kernel. When choosing both K and L_w with BIC, SLLiM achieves the same prediction rate. However, SLLiM performs better than SVM when K is fixed to 10 and L_w chosen by BIC. RVM, SIR, MARS and random forests are not competitive on this example.

Figure 2 presents the adjustment quality for SVM (linear kernel), SLLiM and GLLiM ($K = 10$, L_w estimated using BIC). The first row shows the predicted sucrose levels against the true ones and the second row shows quantile vs quantile plots (QQ-plots) of the predicted sucrose levels as a function of the observed ones. These plots illustrate graphically that SLLiM achieves the best adjustment of the observed responses in particular compare to SVM which generates a number of very bad predictions (Figure 2 (c)).

6.2. Hyperspectral data from Mars

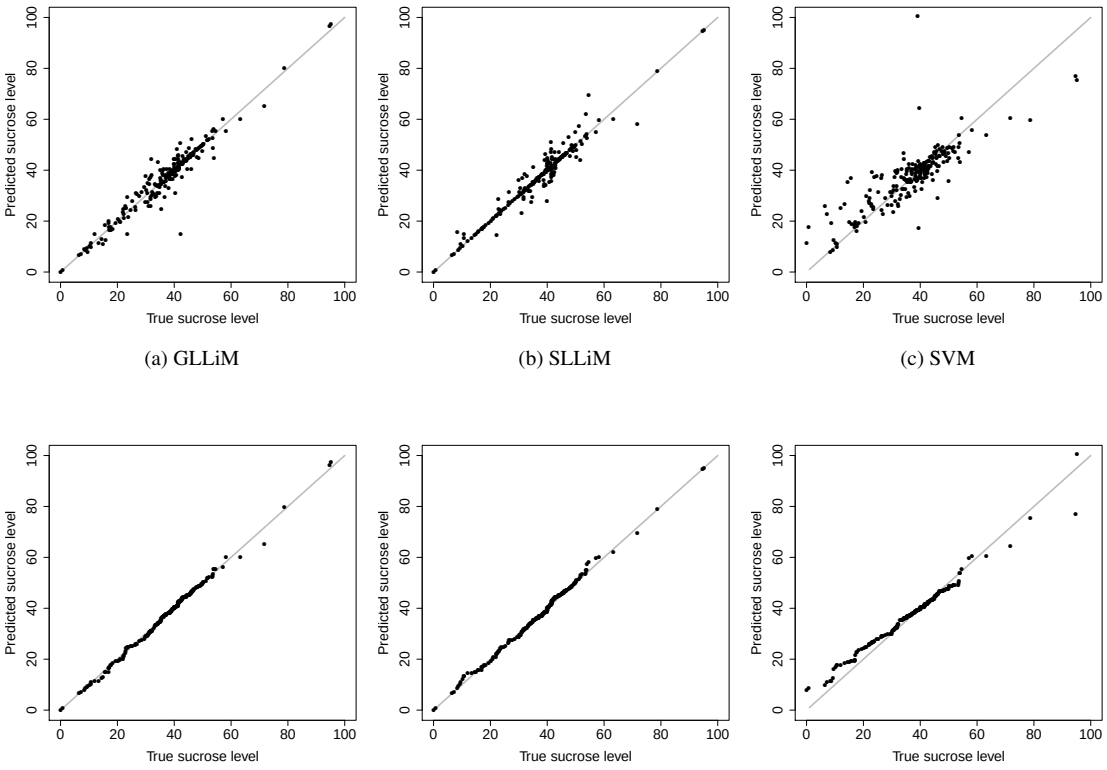
We now investigate the effect of additional robustness on a dataset already studied in [11] and for which good results were already observed with GLLiM.

Data. As described into more details in [11], the dataset corresponds to hyperspectral images of the planet Mars. Spectra are acquired at several locations on the planet and the goal is to recover from each spectrum some physical characteristics of the surface at each location. To do so, a training dataset is available made of $N = 6983$ spectra of length $D = 184$ synthesized from surface characteristics using a radiative transfer model designed by experts. A testing dataset is then available corresponding to spectra acquired on the south polar cap of Mars and for which the true surface characteristics are not known. More specifically, we focus on recovering two quantities, the proportion of CO_2 ice and the proportion of dust. Our observed response variable is therefore bivariate with $L_t = 2$.

Method. The same methods as in the previous section are compared, except SVM and random forests which cannot handle multivariate responses. For GLLiM and SLLiM, the number of clusters K is estimated by BIC or fixed to $K = 10$. A smaller value of K was chosen compare to [11] for several reasons. We observe that for larger values of K the likelihood exhibits some noisy behavior and is not regularly increasing as it should. We therefore suspect that the number of training data may be too small when the number of parameters increases. Then, as mentioned earlier, we are rather interested in investigating the choice of L_w and for the previous reason, it may not be reliable to both increase K and L_w considering the available sample size in this example. The number of additional latent variables L_w is chosen using BIC. The prediction accuracy is first evaluated on the training set using the NRMSE and a cross-validation setting. 100 datasets of size 6000 are randomly sampled among the $N = 6983$ observations and NRMSE are computed by predicting simultaneously the proportions of dust and CO_2 ice on the 983 remaining observations.

Results on training data. Table 3 presents the prediction accuracy achieved by the different tested methods. For SIR, best prediction rates are achieved for 10 directions. For RVM, optimal results are obtained with a Cauchy kernel (heavy-tailed kernel) and a scale parameter set to 1. SLLiM performs better predictions than GLLiM. Among other methods, regression splines (MARS) achieves the best predictions but slightly worse than SLLiM. SIR achieves good

Figure 2: Adjustment on training data



(a) GLLiM (b) SLLiM (c) SVM

(d) GLLiM-QQ-plot (e) SLLiM-QQ-plot (f) SVM-QQ-plot

prediction rates for the proportion of dust but not for the proportion of CO_2 ice and RVM provides the worst results in this example.

Application to Mars surface properties retrieval from hyperspectral images. The training of the radiative model database can then be used to predict proportions of interest from real observed spectra acquired as images. In particular, we focus on a dataset of Mars South polar cap corresponding to a 128×265 image [4]. Since no ground truth is currently available for the physical properties of Mars polar regions, we propose a qualitative evaluation using the three best performing methods among the tested ones, namely SLLiM and GLLiM with $K = 10$ and MARS. All methods appear to match satisfyingly the expected results from planetology experts. Indeed, the observed region is expected to be made of CO_2 ice with increasing amount of dust at the borders with non icy regions. All retrieved images show satisfyingly this dust proportion variation. The main difference between the methods lies in the proportions ranges. SLLiM provides CO_2 ice proportions much higher in the central part of the cap, while MARS provides smoother values all over the cap. According to SLLiM the CO_2 ice would be purer with almost no dust in the central part.

7. Conclusion

Experiments on synthetic and real world data have been conducted to illustrate the empirical usefulness of the proposed method. In practice, real applications raise the issue of making appropriate choices for the number of linear regressions and the number of latent variables. We proposed to use a standard BIC to deal with this issue with

Table 3: Mars data: average NRMSE and standard deviations in parenthesis for proportions of CO_2 ice and dust over 100 runs.

Method	Prop. of CO_2 ice	Prop. of dust
SLLiM (BIC)	0.258 (0.035)	0.257 (0.043)
SLLiM (K=10)	**0.168 (0.019)**	**0.145 (0.020)**
GLLiM (BIC)	0.197 (0.024)	0.173 (0.022)
GLLiM (K=10)	**0.180 (0.023)**	**0.155 (0.023)**
MARS	**0.173 (0.016)**	**0.160 (0.021)**
SIR	0.243 (0.025)	0.157 (0.016)
RVM	0.299 (0.021)	0.275 (0.034)

satisfying results but this is one aspect that could be further investigated, in particular in a context where the number of available training data may not be large enough for selection criteria to be theoretically reliable.

In this paper the main target was to address the fact that an outlier may distort the derivation of the linear mappings so as to fit the outlier well, and therefore may result in wrong parameter estimation. Outlier contamination may then generate distortion during model building and in subsequent prediction. While we could indeed check on low-dimensional synthetic examples that our modeling was effective in dealing with this issue, in actual high-dimensional examples it is often only possible to check the better adjustment of our model in terms of prediction errors. The concept of outlier in a high-dimensional space is not obvious and it would be interesting to investigate the use of our model for actual outlier detection, examining for instance the intermediate weight variables computed in the EM algorithm.

More generally, beyond out-of-sample evaluation such as cross-validation checks of graphical QQ plots, other model checking tools could be proposed. The log-likelihood of the fitted model for the observed data can be seen as a measure of overall model fit and is actually already included in our BIC computation but rather used for model selection, that is for scoring of a discrete set of possible models. Other information metrics could certainly be considered by further exploiting the availability of a tractable log-density for the fitted parametric model. For instance, a visual diagnostic of model fit can be derived from the procedure described in [32]. This procedure mimics a Kolmogorov–Smirnov (KS) goodness-of-fit test. It avoids the problematic computation of a multivariate KS statistic by considering the fitted model log-density $\ln p(\cdot;\boldsymbol{\theta})$ and by comparing the two distributions of the random variable $\ln p(\mathbf{X},\mathbf{Y};\boldsymbol{\theta})$ when $(\mathbf{X},\mathbf{Y})$ follows the fitted $p(\cdot;\boldsymbol{\theta})$ and the true distribution, both approximated using their empirical estimation. This however remains quite heuristic and should be compared to other multivariate KS procedures.

As another future work, our derivations would be similar for other members in the scale mixture of Gaussians family or among other elliptical distributions. It would be interesting to study in a regression context, the use of distributions with even more flexible tails such as multiple scale Student distributions [15] or various skew-t [23, 26, 28] or Normal Inverse Gaussian distributions and more generally non-elliptically contoured distributions [32, 41]. At last, another interesting direction of research would be to further complement the model with sparsity inducing penalties in particular for situations where interpreting the influential covariates is important.

8. Acknowledgments

This work has received support from a XEROX University Affairs Committee (UAC) grant (2015–17).

Appendix A. EM algorithm

Appendix A.1. Expectation step

The expectation step splits into three steps in which the distributions of $\widetilde{r}_Z$, $\widetilde{r}_{U|Z}$ and $\widetilde{r}_{W|Z,U}$ are specified. Some of the computation is similar to that of the Gaussian case and details can be found in [11].

E-W step. The distribution $\widetilde{r}_{W|Z,U}$ is fully specified by computing the $N \times K$ functions $p(\mathbf{w}_n \mid \mathbf{x}_n, \mathbf{t}_n, Z_n = k, U_n = u_n; \boldsymbol{\theta}^{(i)})$ which are all Gaussian distribution functions with expectation $\widetilde{\boldsymbol{\mu}}_{nk}^w$ and variance $\widetilde{\boldsymbol{S}}_k^w/u_n$ defined as:

$$\widetilde{\boldsymbol{\mu}}_{nk}^w = \widetilde{\boldsymbol{S}}_k^w\{\boldsymbol{A}_k^{w(i)\top}\boldsymbol{\Sigma}_k^{(i)-1}(\mathbf{x}_n - \boldsymbol{A}_k^{t(i)}\mathbf{t}_n - \boldsymbol{b}_k^{(i)}) + \boldsymbol{\Gamma}_k^{w(i)-1}\boldsymbol{c}_k^{w(i)}\}, \quad \widetilde{\boldsymbol{S}}_k^w = (\boldsymbol{\Gamma}_k^{w(i)-1} + \boldsymbol{A}_k^{w(i)\top}\boldsymbol{\Sigma}_k^{(i)-1}\boldsymbol{A}_k^{w(i)})^{-1}.$$

E-U step. Similarly, $\widetilde{r}_{U|Z}$ is fully defined by computing for $n = 1 : N$ and $k = 1 : K$, the distribution $p(u_n \mid \mathbf{x}_n, \mathbf{t}_n, Z_n = k; \boldsymbol{\theta}^{(i)})$, which is the density function of a Gamma distribution $\mathcal{G}(u_n, \alpha_k^{t(i)}, \gamma_k^{t(i)})$ with parameters:

$$\alpha_k^{t(i+1)} = \alpha_k^{(i)} + \frac{L_t + D}{2},$$
$$\gamma_k^{t(i+1)} = 1 + \frac{1}{2}\{\delta(\mathbf{x}_n, \boldsymbol{A}_k^{(i)}[\mathbf{t}_n, \boldsymbol{c}_k^{w(i)}]^\top + \boldsymbol{b}_k^{(i)}, \boldsymbol{\Sigma}_k^{(i)} + \boldsymbol{A}_k^{w(i)}\boldsymbol{\Gamma}_k^{w(i)}\boldsymbol{A}_k^{w(i)\top}) + \delta(\mathbf{t}_n, \boldsymbol{c}_k^{t(i)}, \boldsymbol{\Gamma}_k^{t(i)})\}.$$

The values $\alpha_k^{t(i+1)}$ and $\gamma_k^{t(i+1)}$ are deduced using that $p(u_n \mid \mathbf{x}_n, \mathbf{t}_n, Z_n = k; \theta^{(i)})$ is proportional to $p(\mathbf{x}_n \mid \mathbf{t}_n, Z_n = k, U_n = u_n; \boldsymbol{\theta}^{(i)}) p(\mathbf{t}_n \mid Z_n = k, U_n = u_n; \boldsymbol{\theta}^{(i)}) p(u_n | Z_n = k; \boldsymbol{\theta}^{(i)})$ and by noticing that the Gamma distribution is a conjuguate prior for a Gaussian likelihood. As in traditional Student mixtures (see, e.g., [33]), the E-$\boldsymbol{U}$ step actually reduces to the computation of the conditional expectation

$$\begin{aligned}\bar{u}_{nk}^{(i+1)} &= \mathrm{E}(U_n \mid \mathbf{t}_n, \mathbf{x}_n, Z_n = k; \boldsymbol{\theta}^{(i)}) = \frac{\alpha_k^{t(i+1)}}{\gamma_k^{t(i+1)}} \\ &= \frac{\alpha_k^{(i)} + (L_t + D)/2}{1 + 1/2\{\delta(\mathbf{x}_n, \boldsymbol{A}_k^{(i)}[\mathbf{t}_n, \boldsymbol{c}_k^{w(i)}]^\top + \boldsymbol{b}_k^{(i)}, \boldsymbol{\Sigma}_k^{(i)} + \boldsymbol{A}_k^{w(i)}\boldsymbol{\Gamma}_k^{w(i)}\boldsymbol{A}_k^{w(i)\top}) + \delta(\mathbf{t}_n, \boldsymbol{c}_k^{t(i)}, \boldsymbol{\Gamma}_k^{t(i)})\}}.\end{aligned}$$

When $\mathbf{x}_n$ gets away from $\boldsymbol{A}_k^{(i)}\mathbf{y}_n + \boldsymbol{b}_k^{(i)}$ or $\mathbf{t}_n$ from $\boldsymbol{c}_k^{t(i)}$ or both, then the Mahalanobis distances in the denominator increase and $\bar{u}_{nk}^{(i+1)}$ decreases. $\bar{u}_{nk}^{(i+1)}$ acts as a weight. A low $\bar{u}_{nk}^{(i+1)}$ downweights the impact of $\mathbf{t}_n$ and $\mathbf{x}_n$ in the parameters estimations (see below). In the following, the covariance matrix $\boldsymbol{\Sigma}_k^{(i)} + \boldsymbol{A}_k^{w(i)}\boldsymbol{\Gamma}_k^{w(i)}\boldsymbol{A}_k^{w(i)\top}$ is denoted by $\widetilde{\boldsymbol{S}}_k^u$.

E-Z step. Characterizing $\widetilde{r}_Z$ is equivalent to compute each $r_{nk}^{(i+1)}$ defined as the posterior probability that $(\mathbf{t}_n, \mathbf{x}_n)$ belongs to the kth component of the mixture given the current estimates of the mixture parameters $\boldsymbol{\theta}^{(i)}$, viz.

$$r_{nk}^{(i+1)} = \frac{\pi_k^{(i)} p(\mathbf{t}_n, \mathbf{x}_n | Z_n = k; \boldsymbol{\theta}^{(i)})}{\sum_{j=1}^{K} \pi_j^{(i)} p(\mathbf{t}_n, \mathbf{x}_n | Z_n = j; \boldsymbol{\theta}^{(i)})},$$

where the joint distribution $p(t_n, y_n \mid Z_n = k; \boldsymbol{\theta}^{(i)})$ is an $(L_t + D)$-dimensional generalized Student distribution denoted by $\mathcal{S}_{L_t+D}([\mathbf{t}_n, \mathbf{x}_n]^\top; \boldsymbol{m}_k^{t(i)}, \boldsymbol{V}_k^{t(i)}, \alpha_k^{(i)}, 1)$ with $\boldsymbol{m}_k^{t(i)}$ and $\boldsymbol{V}_k^{t(i)}$ defined as

$$\boldsymbol{m}_k^{t(i)} = \begin{bmatrix} \boldsymbol{c}_k^{t(i)} \\ \boldsymbol{A}_k^{(i)}\boldsymbol{c}_k^{(i)} + \boldsymbol{b}_k^{(i)} \end{bmatrix}, \quad \boldsymbol{V}_k^{t(i)} = \begin{bmatrix} \boldsymbol{\Gamma}_k^{t(i)} & \boldsymbol{\Gamma}_k^{t(i)}\boldsymbol{A}_k^{t(i)\top} \\ \boldsymbol{A}_k^{t(i)}\boldsymbol{\Gamma}_k^{t(i)} & \boldsymbol{\Sigma}_k^{(i)} + \boldsymbol{A}_k^{(i)}\boldsymbol{\Gamma}_k^{(i)}\boldsymbol{A}_k^{(i)\top} \end{bmatrix}.$$

Appendix A.2. Maximization step

The update of the parameters decomposes into three parts. Update equations for $\{\pi_k, \boldsymbol{c}_k, \boldsymbol{\Gamma}_k, \alpha_k\}$ are derived from the second part of the expected log-likelihood in Exp. (15) and can be straightforwardly derived from previous work on Student mixtures; see, e.g., [33]. The update of $\{\boldsymbol{A}_k, \boldsymbol{b}_k, \boldsymbol{\Sigma}_k\}$ is deduced from the first part of Exp. (15) and generalizes the corresponding step in [11]. The Student case involves classically some *double weights* accounting for the introduction of an extra latent variable U: $\widetilde{r}_{nk}^{(i+1)} = r_{nk}^{(i+1)}\bar{u}_{nk}^{(i+1)}$. We use also the following notation:

$$\widetilde{r}_k^{(i+1)} = \sum_{n=1}^{N} \widetilde{r}_{nk}^{(i+1)}, \quad r_k^{(i+1)} = \sum_{n=1}^{N} r_{nk}^{(i+1)}.$$

M-$(\pi_k, \boldsymbol{c}_k, \boldsymbol{\Gamma}_k)$ *step.* We recover the Student mixture formula. For this part the model behaves as a Student mixture on the $\{\mathbf{t}_n\}_{n=1}^N$, which gives the following updates:

$$\pi_k^{(i+1)} = \frac{r_k^{(i+1)}}{N}, \quad \boldsymbol{c}_k^{t(i+1)} = \sum_{n=1}^{N} \frac{\widetilde{r}_{kn}^{(i+1)}}{\widetilde{r}_k^{(i+1)}} \mathbf{t}_n, \quad \boldsymbol{\Gamma}_k^{(i+1)} = \sum_{n=1}^{N} \frac{\widetilde{r}_{kn}^{(i+1)}}{r_k^{(i+1)}} (\mathbf{t}_n - \boldsymbol{c}_k^{t(i+1)})(\mathbf{t}_n - \boldsymbol{c}_k^{t(i+1)})^\top$$

M-α_k step. The estimates do not exist in closed form, but can be computed by setting the following expression to 0 (see [15] for details):

$$-\Upsilon(\alpha_k) + \Upsilon\left(\alpha_k^{(i)} + \frac{L_t + D}{2}\right) - \frac{1}{r_k^{(i+1)}} \sum_{n=1}^{N} r_{nk}^{(i+1)} \ln\left[1 + \frac{1}{2}\left\{\delta(\mathbf{x}_n, \boldsymbol{A}_k^{(i)}[\mathbf{t}_n, \boldsymbol{c}_k^{w(i)}]^\top + \boldsymbol{b}_k^{(i)}, \widetilde{\boldsymbol{S}}_k^u) + \delta(\mathbf{t}_n, \boldsymbol{c}_k^{t(i)}, \boldsymbol{\Gamma}_k^{t(i)})\right\}\right]$$

which gives that α_k is estimated by numerically computing:

$$\alpha_k^{(i+1)} = \Upsilon^{-1}\left[\Upsilon\left(\alpha_k^{(i)} + \frac{L_t + D}{2}\right) - \frac{1}{r_k^{(i+1)}} \sum_{n=1}^{N} r_{nk}^{(i+1)} \ln\left[1 + \frac{1}{2}\left\{\delta(\mathbf{x}_n, \boldsymbol{A}_k^{(i)}[\mathbf{t}_n, \boldsymbol{c}_k^{w(i)}]^\top + \boldsymbol{b}_k^{(i)}, \widetilde{\boldsymbol{S}}_k^u) + \delta(\boldsymbol{t}_n, \boldsymbol{c}_k^{t(i)}, \boldsymbol{\Gamma}_k^{t(i)})\right\}\right]\right],$$

where Υ is the Digamma function that verifies $\mathrm{E}(\ln W) = \Upsilon(\alpha) - \ln\gamma$ when W follows a $\mathcal{G}(\alpha, \gamma)$ distribution. The Digamma function also satisfies $d\ln\{\Gamma(\alpha)\}/d\alpha = \Upsilon(\alpha)$.

M-$(\boldsymbol{A}_k, \boldsymbol{b}_k, \boldsymbol{\Sigma}_k)$ step. The updating of the mapping parameters $\{\boldsymbol{A}_k, \boldsymbol{b}_k, \boldsymbol{\Sigma}_k\}_{k=1}^K$ is also in closed form and is obtained by maximizing the first part in (15). It is easy to see that the results in [11] can be used by replacing $r_{nk}^{(i+1)}$ with $\widetilde{r}_{nk}^{(i+1)}$ to account for the extra $\boldsymbol{U}$ variables. It follows that

$$\begin{aligned}
\boldsymbol{A}_k^{(i+1)} &= \widetilde{\mathbf{X}}_k \widetilde{\mathbf{Y}}_k^\top (\widetilde{\boldsymbol{S}}_k^y + \widetilde{\mathbf{Y}}_k \widetilde{\mathbf{Y}}_k^\top)^{-1}, \\
\boldsymbol{b}_k^{(i+1)} &= \sum_{n=1}^{N} \frac{\widetilde{r}_{kn}^{(i+1)}}{\widetilde{r}_k^{(i+1)}} (\mathbf{x}_n - \boldsymbol{A}_k^{(i+1)} \widetilde{\mathbf{y}}_{nk}), \\
\boldsymbol{\Sigma}_k^{(i+1)} &= \boldsymbol{A}_k^{w(i+1)} \widetilde{\boldsymbol{S}}_k^w \boldsymbol{A}_k^{w(i+1)\top} + \sum_{n=1}^{N} \frac{\widetilde{r}_{kn}^{(i+1)}}{r_k^{(i+1)}} (\mathbf{x}_n - \boldsymbol{A}_k^{(i+1)} \widetilde{\mathbf{y}}_{nk} - \boldsymbol{b}_k^{(i+1)})(\mathbf{x}_n - \boldsymbol{A}_k^{(i+1)} \widetilde{\mathbf{y}}_{nk} - \boldsymbol{b}_k^{(i+1)})^\top,
\end{aligned} \tag{A.1}$$

where $\widetilde{\mathbf{y}}_{nk} = [\mathbf{t}_n, \widetilde{\boldsymbol{\mu}}_{nk}^w]^\top$, $\widetilde{\boldsymbol{S}}_k^y = \begin{bmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \widetilde{\boldsymbol{S}}_k^w \end{bmatrix}$ and

$$\widetilde{\mathbf{X}}_k = \frac{1}{\sqrt{r_k}}\left[\sqrt{\widetilde{r}_{1k}}(\mathbf{x}_1 - \widetilde{\mathbf{x}}_k), \ldots, \sqrt{\widetilde{r}_{Nk}}(\mathbf{x}_N - \widetilde{\mathbf{x}}_k)\right] \quad \text{with } \widetilde{\mathbf{x}}_k = \sum_{n=1}^{N} \frac{\widetilde{r}_{kn}}{r_k} \mathbf{x}_n,$$

$$\widetilde{\mathbf{Y}}_k = \frac{1}{\sqrt{r_k}}\left[\sqrt{\widetilde{r}_{1k}}(\widetilde{\mathbf{y}}_{1k} - \widetilde{\mathbf{y}}_k), \ldots, \sqrt{\widetilde{r}_{Nk}}(\widetilde{\mathbf{y}}_{Nk} - \widetilde{\mathbf{y}}_k)\right] \quad \text{with } \widetilde{\mathbf{y}}_k = \sum_{n=1}^{N} \frac{\widetilde{r}_{kn}}{r_k} \widetilde{\mathbf{y}}_{nk}.$$

Appendix A.3. Constrained estimations

The **E** and **M** steps above are given for general $\boldsymbol{\Sigma}_k$. However in practice, for high D, a great gain in complexity can be achieved by imposing some simplifying constraints on $\boldsymbol{\Sigma}_k$. When $\boldsymbol{\Sigma}_k$ is assumed diagonal, it can be estimated by the diagonal of $\boldsymbol{\Sigma}_k^{(i+1)}$ given by (A.1). In the isotropic case, $\boldsymbol{\Sigma}_k = \sigma_k^2 \mathbb{I}_D$, we just need to compute

$$\sigma_k^{2(i+1)} = \frac{1}{D} \operatorname{trace}(\boldsymbol{\Sigma}_k^{(i+1)}). \tag{A.2}$$

In the isotropic and equal case, $\boldsymbol{\Sigma}_k = \sigma^2 \mathbb{I}_D$ for all k, the unique variance parameter is then updated by $\sigma^{2(i+1)} = \sum_{k=1}^K \pi_k^{(i+1)} \sigma_k^{2(i+1)}$, with the expression (A.2) just above.

Appendix A.4. Initialization

Initial values for the **E-U** and **E-Z** steps are natural: the $\bar{u}_{nk}^{(0)}$'s can be set to 1 while the $r_{nk}^{(0)}$'s can be set to the values obtained with a standard EM algorithm for a K-component Gaussian mixture on $(\mathbf{X}, \mathbf{T})$. Steps that involve $\mathbf{W}$ are less straightforward to initialize. Therefore, one solution is to consider a marginal EM algorithm in which the latent variable $\mathbf{W}$ is integrated out. Considering Exp. (15), one can see that the **E-Z** and **E-U** steps are unchanged and that the **E-W** step is removed. With $\boldsymbol{c}_k^w$ and $\boldsymbol{\Gamma}_k^w$ fixed to $\mathbf{0}_{L_w}$ and $\mathbb{I}_{L_w}$ respectively, the estimation of $(\pi_k, \boldsymbol{c}_k^t, \boldsymbol{\Gamma}_k^t)$ and α_k is unchanged in the M-step. The estimation of $(\boldsymbol{A}_k, \boldsymbol{b}_k, \boldsymbol{\Sigma}_k)$ only involves the observed data $(\mathbf{t}_n)_{n=1:N}$ and can be performed in two steps, a regression step for $(\boldsymbol{A}_k^t, \boldsymbol{b}_k)$ and a PPCA-like step for $(\boldsymbol{A}_k^w, \boldsymbol{\Sigma}_k)$.

M-($\boldsymbol{A}_k^t, \boldsymbol{b}_k$) *step.*

$$\boldsymbol{A}_k^{t(i+1)} = \widetilde{\mathbf{X}}_k\widetilde{\mathbf{T}}_k^\top(\widetilde{\mathbf{T}}_k\widetilde{\mathbf{T}}_k^\top)^{-1}, \quad \boldsymbol{b}_k^{(i+1)} = \sum_{n=1}^{N}\frac{\widetilde{r}_{nk}}{\widetilde{r}_k}(\mathbf{x}_n - \boldsymbol{A}_k^{t(i+1)}\mathbf{t}_n), \tag{A.3}$$

where

$$\widetilde{\mathbf{T}}_k = \left[\frac{\sqrt{\widetilde{r}_{1k}}}{\sqrt{\widetilde{r}_k}}(\mathbf{t}_1 - \widetilde{\mathbf{t}}_k), \ldots, \frac{\sqrt{\widetilde{r}_{Nk}}}{\sqrt{\widetilde{r}_k}}(\mathbf{t}_N - \widetilde{\mathbf{t}}_k)\right], \quad \widetilde{\mathbf{t}}_k = \sum_{n=1}^{N}\frac{\widetilde{r}_{nk}}{\widetilde{r}_k}\mathbf{t}_n$$

M-($\boldsymbol{A}_k^w, \boldsymbol{\Sigma}_k$) *step.* Updates are obtained by minimizing the following criterion:

$$Q_k(\boldsymbol{\Sigma}_k, \boldsymbol{A}_k^w) = \ln|\boldsymbol{\Sigma}_k + \boldsymbol{A}_k^w\boldsymbol{A}_k^{w\top}| + \sum_{n=1}^{N}\frac{\widetilde{r}_{nk}}{r_k}(\mathbf{x}_n - \boldsymbol{A}_k^{t(i+1)}\boldsymbol{t}_n - \boldsymbol{b}_k^{(i+1)})^\top(\boldsymbol{\Sigma}_k + \boldsymbol{A}_k^w\boldsymbol{A}_k^{w\top})^{-1}(\boldsymbol{y}_n - \boldsymbol{A}_k^{t(i+1)}\mathbf{t}_n - \boldsymbol{b}_k^{(i+1)}) .$$

More details on the practical resolution are given in [11]. In practice, only one iteration of this marginal EM is run to initialize the complete EM.

Appendix B. Comparison with a cluster-weighted modeling approach

Although our model was focused on regression aspects, it shares some similarity with the so-called CWtFA clustering technique of [38] that uses factor decompositions of the high-dimensional covariance matrices. To illustrate the difference, we apply SLLiM to the data sets used in [38] which are, however, not high-dimensional: the `f.voles` data from the `Flury` R package and the `UScrime` data from the `MASS` package. The first data set is made of 86 observations divided into two known species of female voles *Microtus californicus* (41 individuals) and *M. ochrogaster* (45 individuals). The goal is to predict age ($L = 1$) on the basis of skull measurements ($D = 6$). The second data set contains aggregate measurements on 47 states of the USA and the goal is to investigate the relationship between the crime rate ($L = 1$) and a number of covariates ($D = 13$). An additional grouping variable is available that indicates the 16 Southern states.

Table B.4 shows the clustering results for CWtFA and SLLiM. As the purpose of this study is to compare the clustering returned by these two methods, we set $K = 2$. We consider a model for SLLiM equivalent to the one considered in [38] with the same assumptions: full covariance matrix for covariates, constant across groups for the `f.voles` data and unconstrained for the `UScrime` data. For the `f.voles` data, as for the CWtFA model, BIC selects one latent factor. The clustering returned by SLLiM is similar to the one returned by CWtFA. On this dataset, the specie appears to be a relevant discriminant variable between individuals as the clustering separates subjects according to this variable (except two errors). For the `UScrime` data, as for CWtFA, BIC selects one latent factor. The clustering returned by SLLiM is different from the one returned by CWtFA which illustrates that SLLiM and CWtFA are not equivalent. Moreover, in contrast to the CWtFA result which finds two clusters, the other methods compared in [38] indicate that the estimated number of clusters is usually larger (three clusters). This suggests that the variable of interest (Southern states) is not the only discriminant variable between states. For example, we suspect differences could exist between East and West states but state labels are not available and results cannot be further analyzed.

References

[1] K.P. Adragni, R.D. Cook, Sufficient dimension reduction and prediction in regression, Phil. Trans. Roy. Soc. A 367 (2009) 4385–4405.
[2] C. Archambeau, M. Verleysen, Robust Bayesian clustering, Neural Networks 20 (2007) 129–138.
[3] J. Bæk, G.J. McLachlan, L.K. Flack, Mixtures of factor analyzers with common factor loadings: Applications to the clustering and visualization of high-dimensional data, IEEE Transactions on Pattern Analysis and Machine Intelligence 32 (2010) 1298–1309.
[4] C. Bernard-Michel, S. Douté, M. Fauvel, L. Gardes, S. Girard, Retrieval of Mars surface physical properties from OMEGA hyperspectral images using Regularized Sliced Inverse Regression, J. Geophysical Res. Planets 114 (E6).
[5] C.M. Bishop, M. Svensen, Robust Bayesian mixture modelling, Neurocomputing 64 (2005) 235–252.
[6] C. Bouveyron, S. Girard, C. Schmid, High dimensional data clustering, Comput. Statist. Data Anal. 52 (2007) 502–519.
[7] L. Breiman, Random forests, Machine Learning 45 (2001) 5–32.

Table B.4: SLLiM and CWtFA clustering results for `f.voles` (a) and `UScrime` (b) data sets. The goal is to assess how the two estimated clusters (columns) fit the two known ones (lines).

		Cluster	
		1	2
CWtFA-CCCU	M. Ochrogaster	43	2
	M. Californicus	0	41
SLLiM-K = 2 – Lw = 1	M. Ochrogaster	43	2
	M. Californicus	0	41

(a) Clustering on `f.voles` dataset library `Flury`

		Cluster	
		1	2
CWtFA-UUUU	Not Southern	30	1
	Southern	3	13
SLLiM-K = 2 – Lw = 1	Not Southern	9	22
	Southern	0	16

(b) Clustering on `UScrime` dataset library `MASS`

[8] F. Chamroukhi, Non-Normal Mixtures of Experts, ArXiv e-prints.
[9] D. Cook, Fisher Lecture: Dimension reduction in regression, Statist. Science 22 (2007) 1–26.
[10] R.D. de Veaux, Mixtures of linear regressions, Comput. Statist. Data Anal. 8 (1989) 227–245.
[11] A. Deleforge, F. Forbes, R. Horaud, High-dimensional regression with Gaussian mixtures and partially-latent response variables, Statist. Computing 25 (2015) 893–911.
[12] E. Devijver, Finite mixture regression: A sparse variable selection by model selection for clustering, Electronic J. Statist. 9 (2015) 2642–2674.
[13] P. Ding, Bayesian robust inference of sample selection using selection-t models, J. Multivariate Anal. (2014) 451–464.
[14] P. Ding, On the conditional distribution of the multivariate t distribution, Amer. Statist. 70 (2016) 293–295.
[15] F. Forbes, D. Wraith, A new family of multivariate heavy-tailed distributions with variable marginal amounts of tailweights: Application to robust clustering, Statist. Comput. 24 (2014) 971–984.
[16] J. Friedman, Multivariate adaptive regression splines (with discussion), Ann. Statist. 19 (1991) 1–141.
[17] S. Frühwirth-Schnatter, Finite Mixture and Markov Switching Models, Springer, New York, 2006.
[18] L.A. García-Escudero, A. Gordaliza, F. Greselin, S. Ingrassia, A. Mayo-Iscar, Robust estimation of mixtures of regressions with random covariates, via trimming and constraints, Statist. Comput. 27 (2017) 377–402.
[19] N. Gershenfeld, Nonlinear Inference and Cluster-Weighted Modeling, Annals of the New York Academy of Sciences 808 (1997) 18–24.
[20] S.M. Goldfeld, R. E. Quandt, A Markov model for switching regressions, J. Econometrics 1 (1973) 3 – 15.
[21] C. Hennig, Identifiability of models for clusterwise linear regression, J. Classif. 17 (2000) 273–296.
[22] S. Ingrassia, S.C. Minotti, G. Vittadini, Local statistical modeling via a cluster-weighted approach with elliptical distributions, J. Classif. 29 (2012) 363–401.
[23] Z. Jiang, P. Ding, Robust modeling using non-elliptically contoured multivariate distributions, J. Statist.Plann. Inf. 177 (2016) 50–63.
[24] A. Karatzoglou, D. Meyer, K. Hornik, Support vector machines in R, J. Statist. Software 15 (2006) 1–28.
[25] S. Kotz, S. Nadarajah, Multivariate t Distributions and Their Applications, Cambridge University press, 2004.
[26] S. Lee, G. McLachlan, Finite mixtures of multivariate skew t-distributions: Some recent and new results, Statist. Comput. 24 (2014) 181–202.
[27] K. Li, Sliced inverse regression for dimension reduction, J. Amer. Statist. Assoc. 86 (1991) 316–327.
[28] T. Lin, Robust mixture modelling using multivariate skew-t distribution, Statist. Comput. 20 (2010) 343–356.
[29] C. Liu, Robit regression: A simple robust alternative to logistic and probit regression, Applied Bayesian Modeling and Causal Inference from Incomplete-data Perspectives (2004) 227–238.
[30] Y.V. Marchenko, M.G. Genton, A Heckman selection t model, J. Amer. Statist. Assoc. 107 (2012) 304–317.
[31] X.-L. Meng, D. Van Dyk, The EM algorithm: An old folk-song sung to a fast new tune, J. Roy. Statist. Soc. Ser. B 59 (1997) 511–567.
[32] A. O'Hagan, T.B. Murphy, I.C. Gormley, P. McNicholas, D. Karlis, Clustering with the multivariate Normal Inverse Gaussian distribution, Comput. Statist. Data Anal. 93 (2016) 18–30.
[33] D. Peel, G. McLachlan, Robust mixture modeling using the t distribution, Statist. Comput. 10 (2000) 339–348.
[34] J.C. Pinheiro, C. Liu, Y.N. Wu, Efficient algorithms for robust estimation in linear mixed-effects models using the multivariate t distribution, J. Comput. Graphical Statist. 10 (2001) 249–276.
[35] R. Rosipal, N. Krämer, Overview and recent advances in partial least squares, in: C. Saunders, M. Grobelnik, S. Gunn, J. Shawe-Taylor (Eds.), Subspace, Latent Structure and Feature Selection, Springer, New York, pp. 34–51, 2006.
[36] N. Städler, P. Bühlmann, S. van de Geer, 1-penalization for mixture regression models, TEST 19 (2010) 209–256.
[37] S. Subedi, A. Punzo, S. Ingrassia, P. McNicholas, Clustering and classification via cluster-weighted factor analyzers, Adv. Data Anal. Classif. 7 (2013) 5–40.
[38] S. Subedi, A. Punzo, S. Ingrassia, P. McNicholas, Cluster-weighted t-factor analyzers for robust model-based clustering and dimension reduction, Statist. Methods Appl. 24 (2015) 623–649.
[39] M.E. Tipping, Sparse Bayesian learning and the relevance vector machine, J. Machine Learning Res. 1 (2001) 211–244.
[40] V. Vapnik, Statistical Learning Theory. Wiley, New York.
[41] D. Wraith, F. Forbes, Location and scale mixtures of Gaussians with flexible tail behaviour: Properties, inference and application to multivariate clustering, Comput. Statist. Data Anal. 90 (2015) 61–73.
[42] H. Wu, Kernel sliced inverse regression with applications to classification, J. Comput. Graphical Statist. 17 (2008) 590–610.
[43] L. Xu, M. Jordan, G. Hinton, An alternative model for mixtures of experts, Adv. Neural Inform. Proc. Systems (1995) 633–640.
[44] W. Yao, Y. Wei, C. Yu, Robust mixture regression using the t-distribution, Comput. Statist. Data Anal. 71 (2014) 116–127.